

Oklahoma Academic Standards for Computer Science {Level 2 Final Public Draft}

Concept	Subconcept	By the end of 12th Grade
Computing Systems	Hardware & Software	L2.CS.HS.01 Identify and categorize roles of an operating system.
	Troubleshooting	L2.CS.T.01 Identify how hardware components facilitate logic, input, output and storage in computing systems.
Networks & The Internet	Network Communication & Organization	L2.NI.NCO.01 Describe the issues that impact network functionality (e.g., bandwidth, load, latency, topology).
	Cybersecurity	L2.NI.C.01 Compare and refine ways in which software developers protect devices and information from unauthorized access.
Data Analysis	Collection, Visualization, & Transformation	L2.DA.CVT.01 Use data analysis tools and techniques to identify patterns from complex real-world data.
		L2.DA.CVT.02 Generate data sets that use a variety of data collection tools and analysis techniques to support a claim and/or communicate information.
	Inference & Models	L2.DA.IM.01 Use models and simulations to help formulate, refine, and test scientific hypotheses.
Algorithms & Programming	Algorithms	L2.AP.A.01 Describe how artificial intelligence algorithms drive many software and physical systems (e.g., autonomous robots, computer vision, pattern recognition, text analysis).
		L2.AP.A.02 Develop an artificial intelligence algorithm to play a game against a human opponent or solve a real-world problem.
		L2.AP.A.03 Critically examine and implement classic algorithms (e.g., selection sort, insertion sort, binary search, linear search).
		L2.AP.A.04 Evaluate algorithms (e.g., sorting, searching) in terms of their efficiency and clarity.
	Variables	L2.AP.V.01 Compare and contrast simple data structures and their uses (e.g., lists, stacks, queues).
	Control	L2.AP.C.01 Trace the execution of iteration (loops or recursion), illustrating output and changes in values of named variables.
	Modularity	L2.AP.M.01 Construct solutions to problems using student-created components (e.g., procedures, modules, objects).
		L2.AP.M.02 Design or redesign a solution to a large-scale computational problem by identifying generalizable patterns.
		L2.AP.M.03 Create programming solutions by reusing existing code (e.g., libraries, Application Programming Interface (APIs), code repositories).
	Program Development	L2.AP.PD.01 Identify and compare features of various programming languages that make them useful for solving problems and developing systems.
		L2.AP.PD.02 Create software that will provide solutions to a variety of users using the software life cycle process.
		L2.AP.PD.03 Design software in a project team environment using integrated development environments (IDEs), versioning systems, and collaboration systems.
		L2.AP.PD.04 Develop programs for multiple computing platforms.
		L2.AP.PD.05 Systematically check code for correctness, usability, readability, efficiency, portability, and scalability through peer review.

Oklahoma Academic Standards for Computer Science {Level 2 Final Public Draft}

Concept	Subconcept	By the end of 12th Grade
		L2.AP.PD.06 Develop and use a series of test cases to verify that a program performs according to its design specifications.
		L2.AP.PD.07 Explain security issues that might lead to compromised computer programs.
		L2.AP.PD.08 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).
Impacts of Computing	Culture	L2.IC.C.01 Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.
		L2.IC.C.02 Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.
		L2.CS.T.01 Identify how hardware components facilitate logic, input, output, and storage in computing systems.
	Safety, Law, & Ethics	L2.IC.SLE.01 Debate laws and regulations that impact the development and use of software.